CERN
CH-1211 Geneva 23
Switzerland

the
Large
Hadron
Collider
project

LHC Project Document No.
LHC-

CERN Div./Group or Supplier/Contractor Document No.
-

EDMS Document No.
-

Date: 2006-09-11

Functional Specification

THE LHC SEQUENCER

Abstract

The following document describes a proposal for the functionality of the high level software applications that will constitute the LHC Sequencer, and its interface with the rest of the systems (collimators, power converters, radio frequency, central timing system (CBCM), etc) needed to operate the LHC in different modes and with different beam types.

Prepared by :	***Checked by :***	
R. Alemany AB/OP **Reyes.Alemany.Fernandez@cern.ch**		

History of Changes

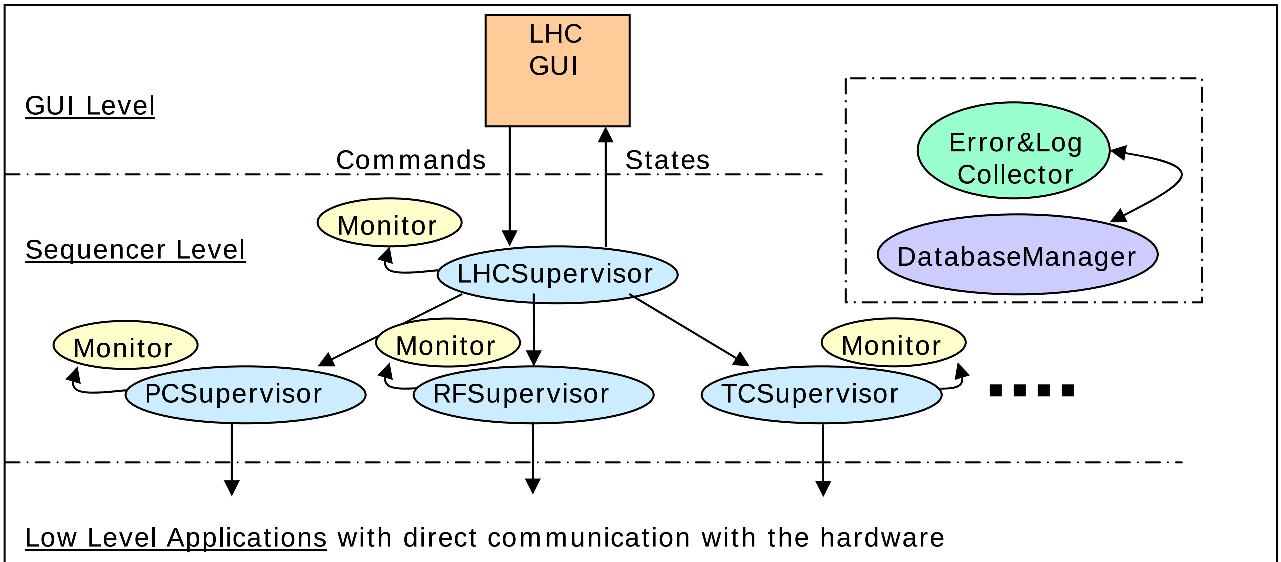
Rev. No.	***Date***	***Pages***	***Description of Changes***
-0.1	2006-09-08		Very preliminary version.

Table of Contents

1. INTRODUCTION

The LHC Sequencer is a set of high level software applications that will operate the LHC accelerator in all its modes and with all kind of beams. These software applications will be based on the AB/CO software tools, including the database. The proposed architecture is organized in five main levels as depicted in Figure 1:

1. The Graphical User Interface (GUI) Level, which has to be simple and handy.
2. The Sequencer Level is made of two sub-levels, the LHCSupervisor and the LHC Systems (Power Converters, RF, Collimation System, BPMs, BLMs, central timing system, other software components, etc). This is a high level software layer that plays the role of the interface between the Operator and the low level applications (including software interlocks) for driving the LHC accelerator.
3. The low level applications controlling the Power Converters, RF, Collimation System, BPMs, BLMs, etc, which are the responsibility of the different groups (I assume): PO, RF, etc.
4. The Error and Log Messages Collector that centralizes and unifies the way the errors, exceptions and log messages are displayed to the Operator on shift. It could also have an error analysis tool.
5. The Database Manager, which is the interface between the Sequencer Level applications and the Database. The writing to and reading from the Database are centralized here.



PCSupervisor: Power Converters System Supervisor
RFSupervisor: Radio Frequence System Supervisor
TCSupervisor: Collimators System and TCDQ Supervisor
MSupervisor: Magnet System (MB,MQ,MS,...) Supervisor
TDSupervisor: Dump for ejected beam (TDE) and beam stopper for injection (TDI)
BISupervisor: Beam Instrumentation System Supervisor (BLMs, BPMs, etc)
CBCMSupervisor: Central Beam and Cycle Manager Supervisor

Figure 1: Architecture of the LHC Sequencer. The LHC Sequencer is composed of a GUI, a LHCSupervisor, a PCSupervisor, a RFSupervisor, a TCSupervisor, a MSupervisor, a TDSupervisor, a BISupervisor, a CBCMSupervisor, an Error and Log Collector and a Database Manager. All are high level software applications.

In Figure 1 the MSupervisor oversees the magnets operation. Similarly, the BISupervisor supervises the operation of the beam instrumentation devices. Nevertheless, depending of the complexity of such a systems, or event just for simple convenience or for performance reasons, the MSupervisor (just to take an example) could be split in different supervisors each one controlling a part of the magnet system.

The LHCSupervisor coordinates the activities of all the systems participating in the running of the selected sequence: power converters, RF, collimators, etc. The PCSupervisor, RFSupervisor, etc, coordinate the activities of the PC, RF, etc, respectively. Therefore, they may have a dedicated FSM different from the LHCSupervisor one. This specialized FSM should interface adequately the LHCSupervisor FSM and the low level applications and the corresponding equipment.

The idea behind this architecture is that the complexity of the LHC operation is factorized and the responsibilities are well defined. This simplification will be also reflected in the software architecture, allowing us to develop software compliant with two fundamental rules: maintainability and scalability (hopefully).

2. THE LHCSUPERVISOR AND THE CONFIGURATION KEYS

The LHCSupervisor is a pure software component that receives from the Operator (via the GUI) the configuration key of the system, i.e. the LHC_SEQUENCE_ID. The LHC_SEQUENCE_ID is the primary key (PK) of a table called LHC_SEQUENCE that contains three foreign keys (FK):

1. The LHC Equipment Key (LHC_EQUIPMENT_ID)
2. The LHC Operation Mode Key (LHC_OPMODE_ID)
3. The LHC Beam Type Key (LHC_BEAMTYPE_ID)

Once a LHC Sequence (LHC_SEQUENCE_ID) is selected by the Operator, the LHCSupervisor will take from the table LHC_SEQUENCE the description of the selected sequence and the corresponding FK for the equipment, beam type and operation mode that corresponds to the selected sequence. The LHC_SEQ-DESCRIPTION is a string easily understandable by the operator and that will be displayed in the GUI.

Figure 2 shows a possible database schema for the LHC_SEQUENCE configuration. The FK will be used in the query to get the corresponding information for the equipment, etc, in the tables LHC-EQUIPMENT, etc.

The LHCSupervisor will parse the information retrieved from the DB to build accordingly the sequence to be played and display the appropriate infor in the GUI and send it to other possible clients.

2.1 THE EQUIPMENT KEY

The LHC_EQUIPMENT_ID is the PK of the LHC_EQUIPMENT table that contains, first of all, an understandable description of the equipment that has to be ready to play the given sequence, and the configuration keys of all the equipment: LHC_PC_ID, LHC_RF_ID, etc. Those are the configuration keys that the LHCSupervisor will send to the PCSupervisor, RFSupervisor, etc, for them to:

1. look for in the corresponding database tables, all the configuration parameters for the power converters and dispatch these configuration data (settings) to the given Power Converters Control Software;
2. or, pass forward the configuration key to the corresponding equipment control software for it to do the job of looking into the database.

(To see what is the best thing to do, it will depend on the system I guess).

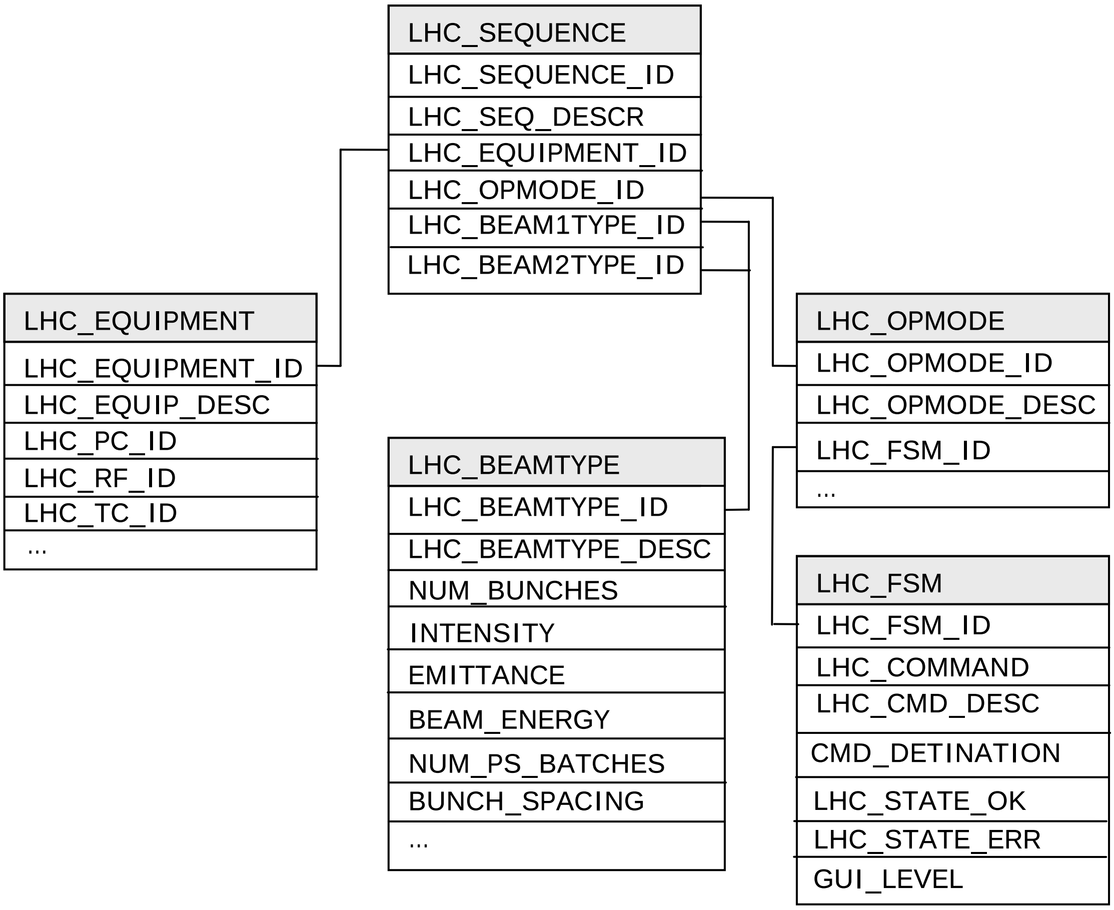


Figure 2: Database schema for the LHC_SEQUENCE, LHC_EQUIPMENT, LHC_BEAMTYPE and LHC_OPMODE tables.

2.2 THE BEAM TYPE KEY

The LHC_BEAMTYPE_ID is the PK of the table LHC_BEAMTYPE containing all the beam definitions [1] needed for the different operation modes. An example of the content of the table is given in Table 1.

Table 1: The LHC_BEAMTYPE table with some examples of LHC_BEAMTYPE_ID.

LHC_BEAMTYPE							
LHC_BEAMTYPE_ID	LHC_BEAMTYPE_DESC	NUM_BUNCHES	NUM_PROT	EMITTANCE	NUM_PS_BATCHES	BUNCH_SPACING	ENERGY
Probe_beam		1	5x10^9	0.8		25	
Single_bunch		1	5x10^9	3.5		25	
Multiple_bunches							
Pilot_beam		1	5x10^9	3.5		25	
Intermediate25		12	1.15 x10^9	3.5		25	
Intermediate75		4	1.15 x10^9	3.5		75	
Physics25		72	1.15 x10^9	3.5		25	
Physics75		24	1.15 x10^9	3.5		75	

2.3 THE OPERATION MODE KEY

The LHC_OPMODE_ID is the PK of the table LHC_OPMODE containing all the possible LHC operation modes: commissioning, machine development, proton physics, ion physics, etc. This table has a FK called LHC_FSM_ID that points to a table called LHC_FSM (LHC Finite State Machine). This last table contains all the commands and states of the LHCSupervisor valid for the selected Sequence. The description of this table is the following:

- Every command is assigned a destination (CMD_DESTINATION column), i.e., the PC(Supervisor), the RF(Supervisor), etc, to which the command will be redirected.
- The list of commands and states will be shown in the GUI. Therefore the LHC GUI will be a dynamic display whose appearance will depend on which Sequence it is being played as depicted in Figure 3 and Figure 5.
- The commands and states have two possible levels (GUI_LEVEL column). Level one means that they will be displayed in the main window like the ones in

Figure 3. Level two means that they will be displayed in a dedicated window that appears when the Operator clicks on “Detailed FSM”, like the example in Figure 4.

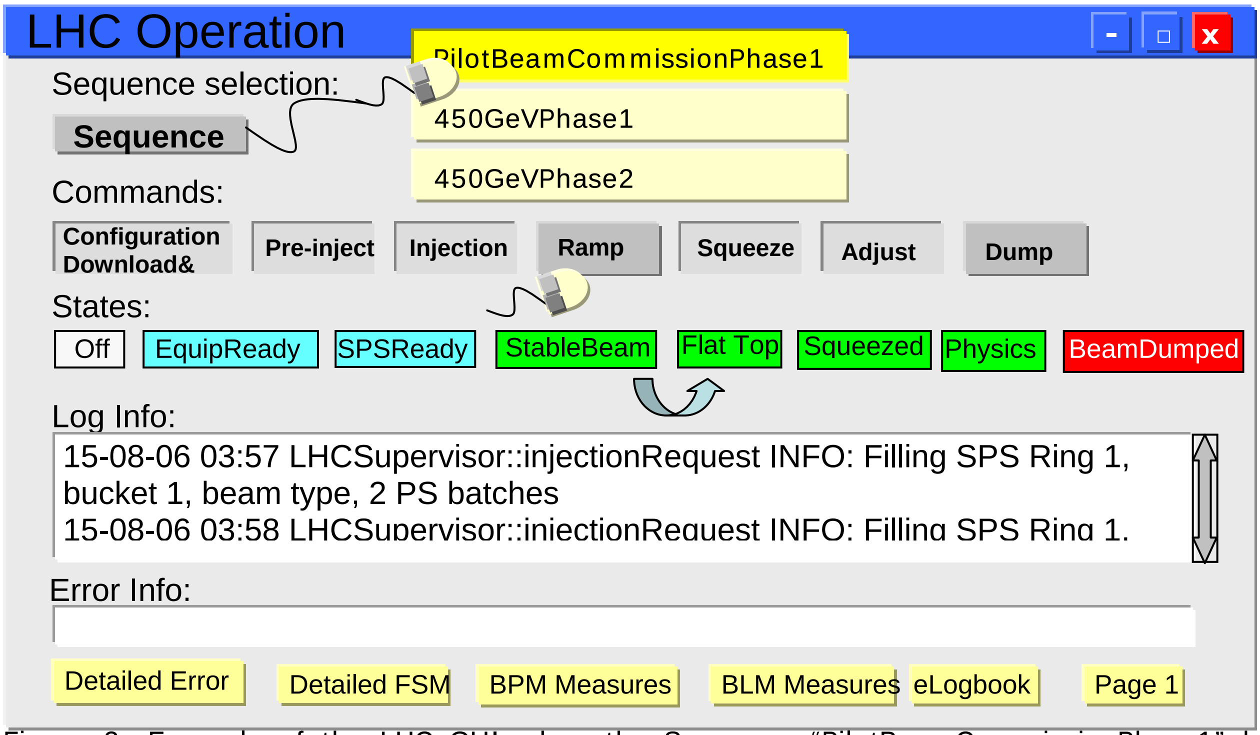
- LHC_STATE_OK and LHC_STATE_ERR are the name of the states the LHCSupervisor can reach after success or failure of the corresponding command, respectively.

Another feature of the GUI could be that only the commands that can be currently issued will appear activated on the window and can be clicked with the mouse. The current state will appear in dark colour, while the other states will appear in light colours.

3. LHCSUPERVISOR STATE MACHINE AND THE GUI

As already said, the LHCSupervisor State Machine will depend on the Sequence is being played. In the following we will discuss the most general State Machine, all the other possible State Machines are a sub-set of this one. The operation of the LHC accelerator can be divided in five main steps or phases:

1. Configuration Parameters Download and Equipment Set up.
2. Pre-Injection
3. Injection
4. Acceleration
5. Dump



```
15-08-06 03:57 LHCSupervisor::injectionRequest INFO: Filling SPS Ring 1,
bucket 1, beam type, 2 PS batches
15-08-06 03:58 LHCSupervisor::injectionRequest INFO: Filling SPS Ring 1.
```

Figure 3: Example of the LHC GUI when the Sequence "PilotBeamCommissionPhase1" has been selected. The Commands and States that will appear on the window will be the ones in the table LHC_FSM for that given sequence which GUI_LEVEL is 1, i.e. they will be displayed on the main window. All the Commands and States with GUI_LEVEL equal to 2, will appear on a dedicated window that will show up when clicking on "Detailed FSM" (something like the FSM shown in Figure 4).

3.1 CONFIGURATION PARAMETERS DOWNLOAD AND EQUIPMENT SET UP

A first draft of the LHCSupervisor State Machine corresponding to this phase is the one represented in Figure 6. As a convention, which will be used in the following intermediate states are in yellow, error and dump states in red, everything related to get the equipment ready is in blue, and finally green for all the states implying circulating beam in the machine (this convention is also used in the GUI). In the figure the actions to go from one state to the other are presented in a sequential manner, however, we could envisage parallel actions when possible.

3.2 PRE-INJECTION

A first draft of the LHCSupervisor State Machine corresponding to the pre-injection phase is the one represented in Figure 7. As can be seen, every command to be sent to the corresponding supervisor, is accompanied by a configuration key to be used by the corresponding supervisor to get the configuration settings from the database for the corresponding equipment.

3.3 INJECTION

A first draft of the LHCSupervisor State Machine corresponding to the injection phase is the one represented in Figure 8.

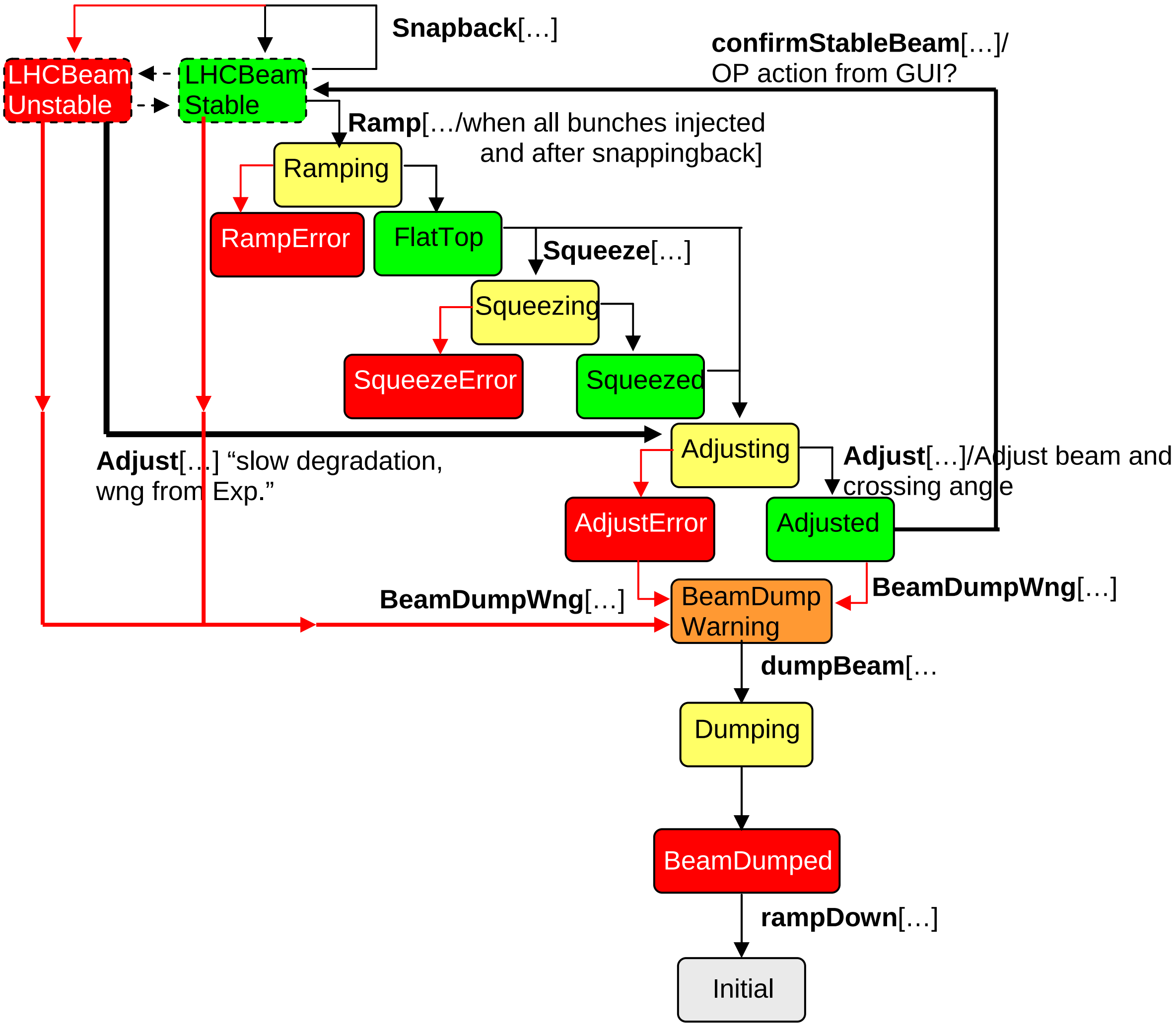


Figure 4: Detailed LHCSupervisor State Machine for the Acceleration step where the commands Ramp, Squeeze, Adjust and BeamDumpWng can be issued.

3.4 ACCELERATION

A first draft of the LHCSupervisor State Machine corresponding to the acceleration phase is the one represented in Figure 9.

3.5 DUMP

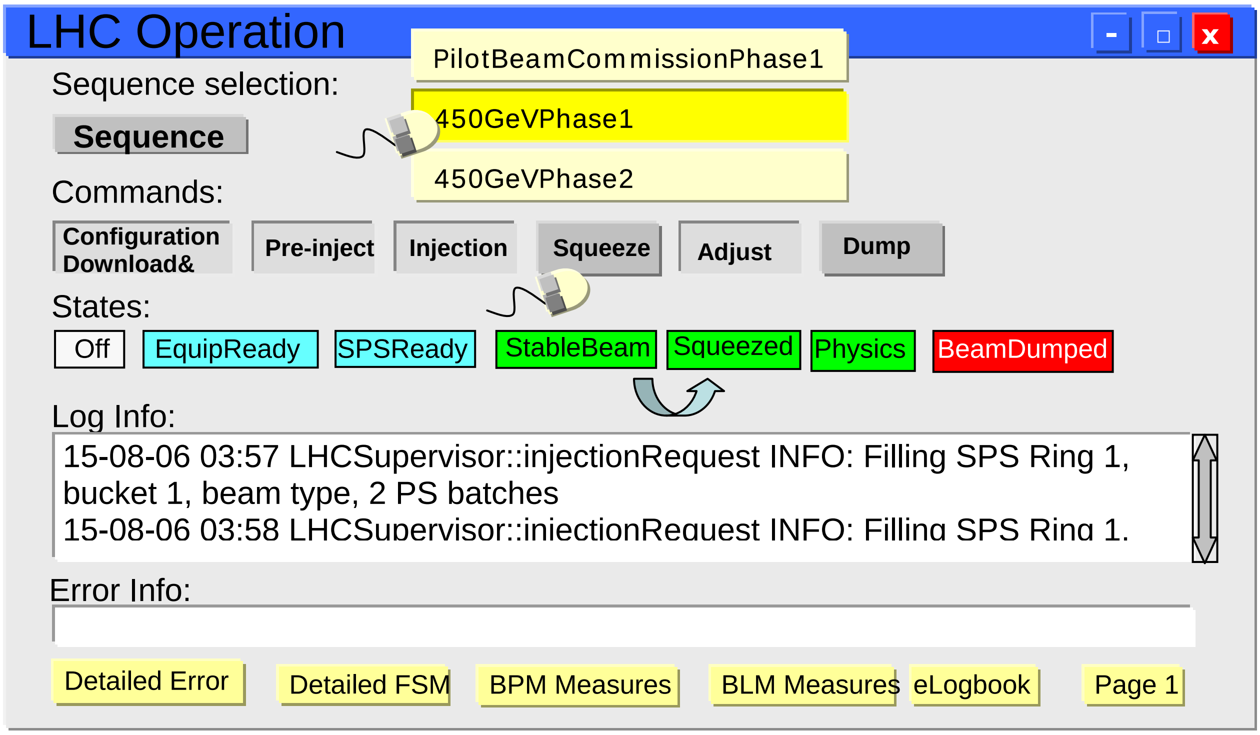
The dump commands and corresponding states are in Figure 9.

4. CONCLUSIONS

This document is a first draft on the functionality of the LHC sequencer. The FSM presented for the different operation phases are a first approximation. They will have to be worked out further.

The interface between the Sequencer and the low level applications controlling the hardware has to be worked out also. This has not been discussed here.

One proposal to see the feasibility of the architecture presented here is the following: implement this schema developing a GUI, the LHCSupervisor and the PCSupervisor with the corresponding interface to the PC controller.



```
15-08-06 03:57 LHCSupervisor::injectionRequest INFO: Filling SPS Ring 1,
bucket 1, beam type, 2 PS batches
15-08-06 03:58 LHCSupervisor::injectionRequest INFO: Filling SPS Ring 1.
```

Figure 5: Example of the LHC GUI when the Sequence “PilotBeamCommissionPhase1” has been selected.

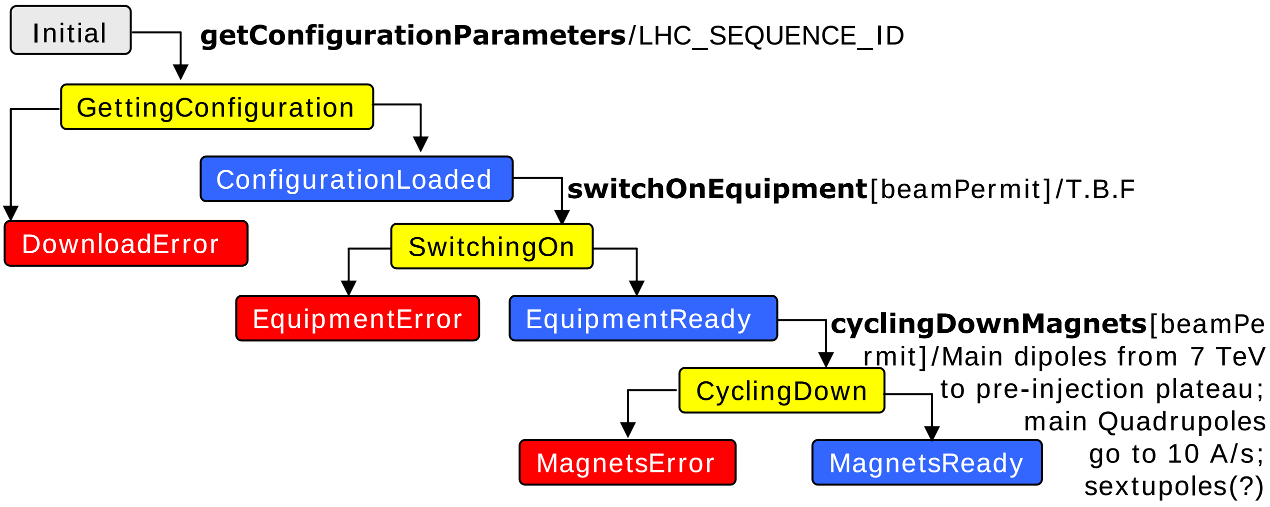

Initial
getConfigurationParameters/LHC_SEQUENCE_ID
GettingConfiguration
ConfigurationLoaded
switchOnEquipment[beamPermit]/T.B.F
DownloadError
SwitchingOn
EquipmentError
EquipmentReady
cyclingDownMagnets[beamPermit]/Main dipoles from 7 TeV to pre-injection plateau; main Quadrupoles go to 10 A/s; sextupoles(?)
CyclingDown
MagnetsError
MagnetsReady

Figure 6: LHCSupervisor State Machine for the “Configuration Parameters Download and Equipment Setup” phase. *To be worked out how to pass the configuration data.*

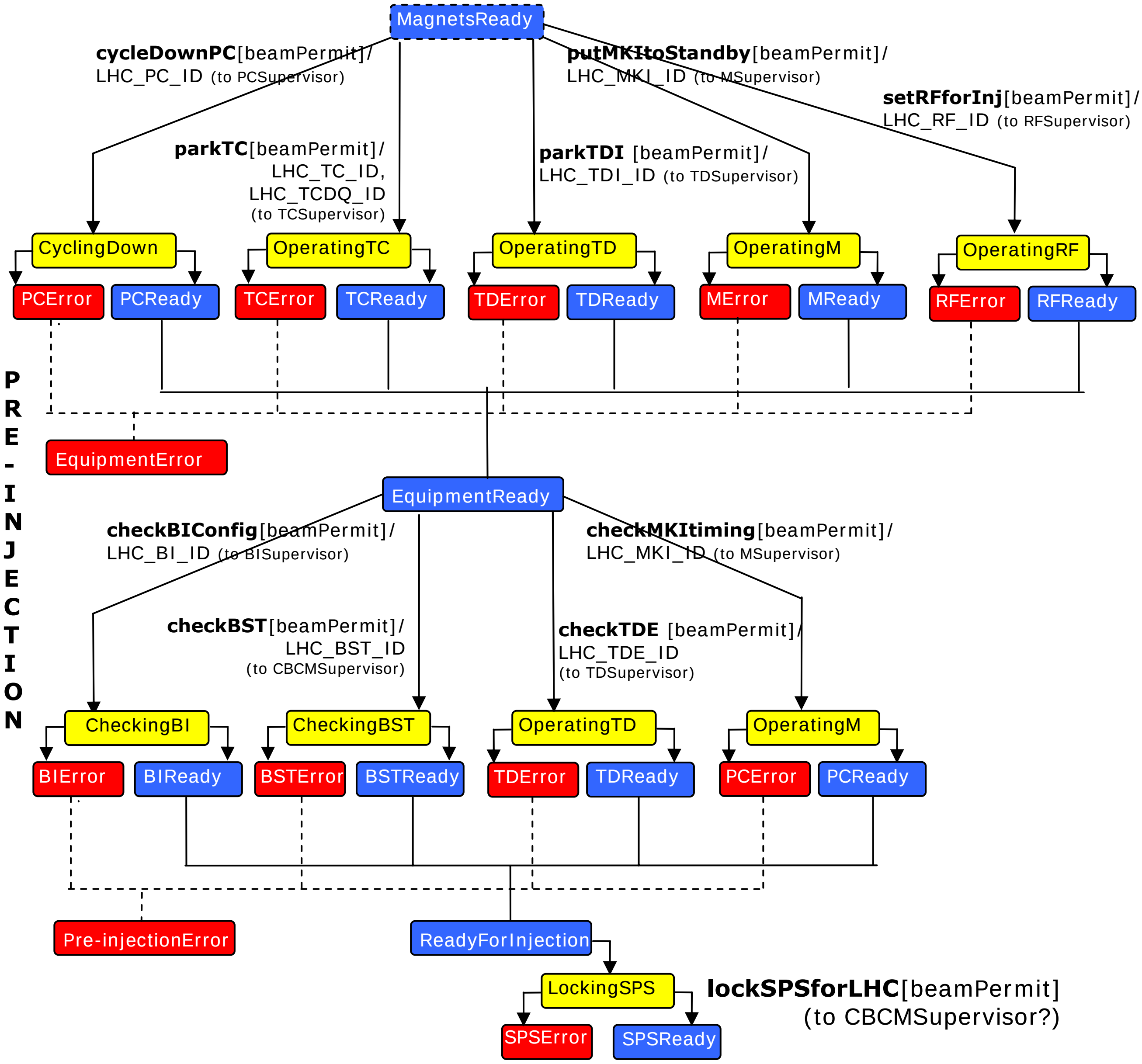


Figure 7: LHCSupervisor State Machine for the “Pre-Injection” phase.

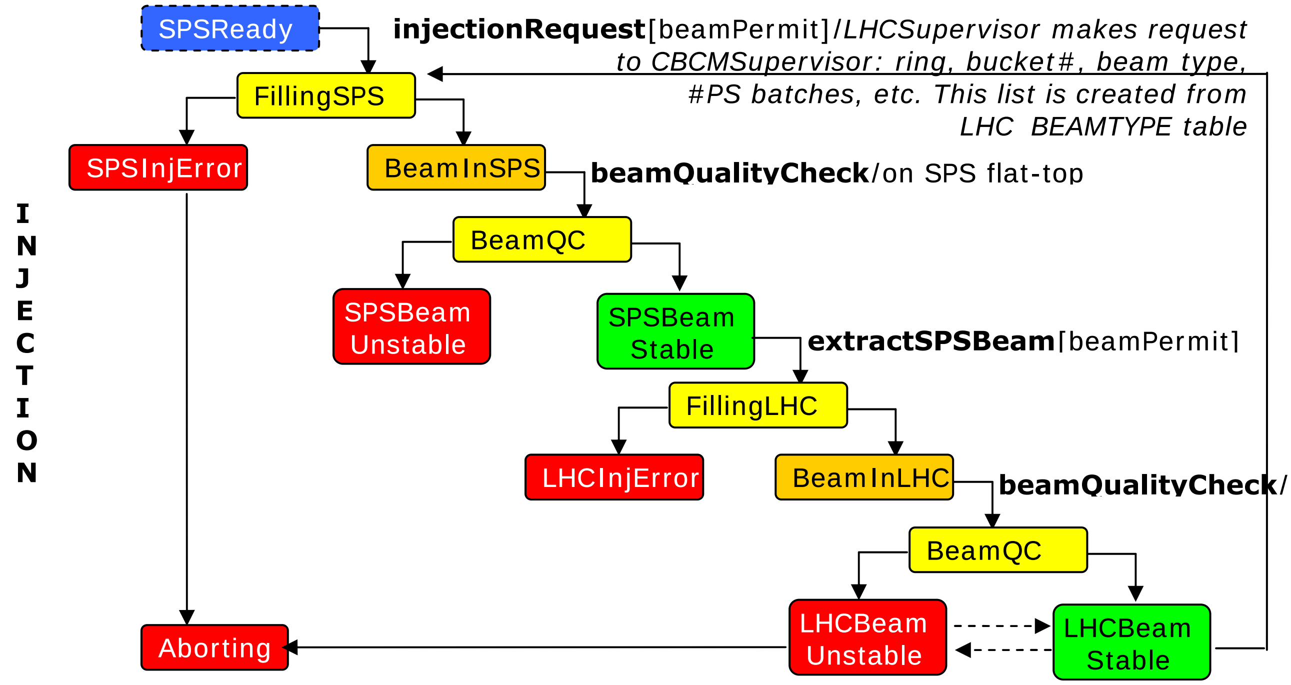

SPSReady
injectionRequest[beamPermit]/LHCSupervisor makes request to CBCMSupervisor: ring, bucket#, beam type, #PS batches, etc. This list is created from LHC BEAMTYPE table
FillingSPS
SPSInjError
BeamInSPS
beamQualityCheck/on SPS flat-top
INJECTION
BeamQC
SPSBeam Unstable
SPSBeam Stable
extractSPSBeam[beamPermit]
FillingLHC
LHCInjError
BeamInLHC
beamQualityCheck/
BeamQC
Aborting
LHCBeam Unstable
LHCBeam Stable

Figure 8: LHCSupervisor State Machine for the “Injection” phase.

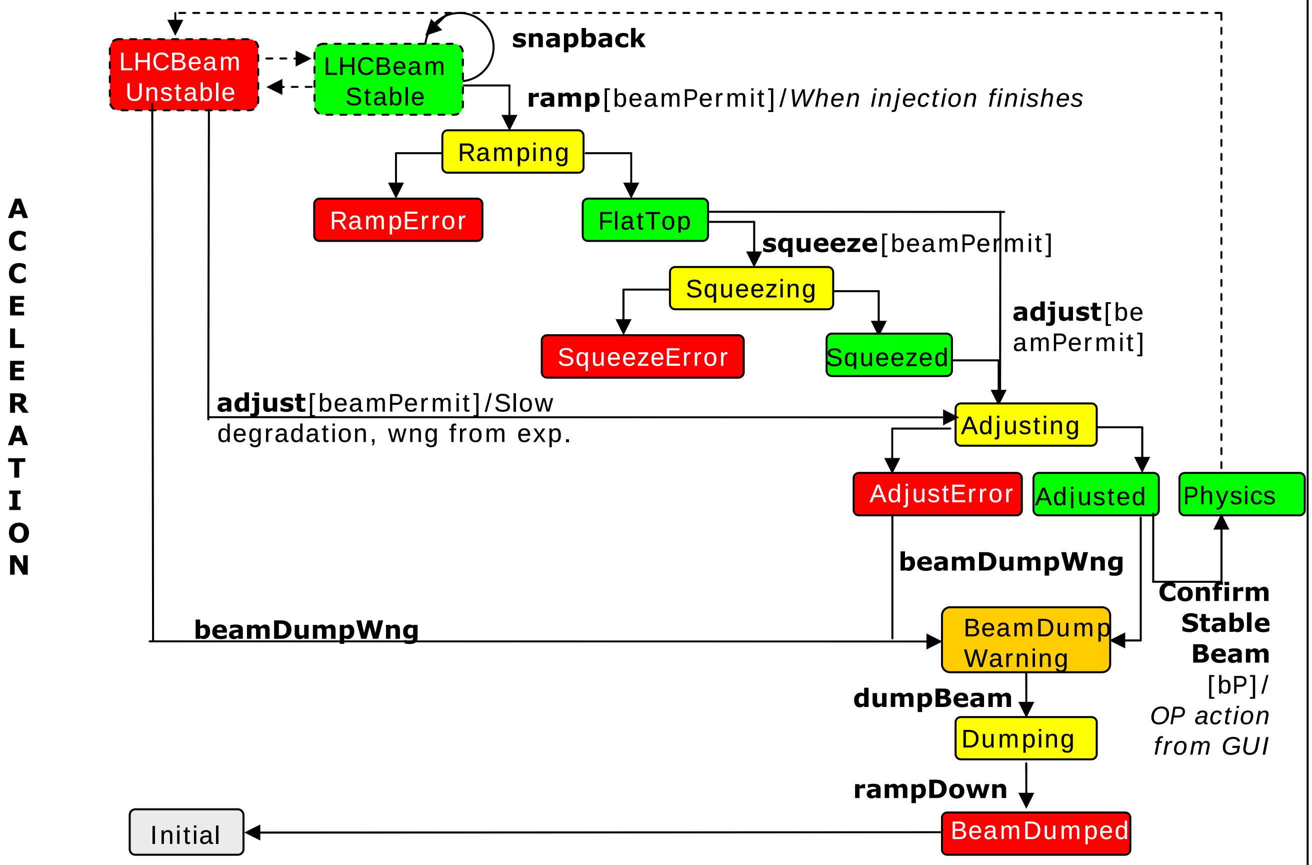


Figure 9: LHCSupervisor State Machine for the “Acceleration” phase.

5. REFERENCES

[1] M. Benedikt, LHC-OP-ES-0002 rev 1.0 (EDMS Doc. No. 487892).